Distributed Design and Distributed Social Awareness: Exploring Inter-subjective Dimensions of Roles

Flore Barcellini, Françoise Détienne, and Jean-Marie Burkhardt

Abstract This research deals with the investigation of inter-subjective dimensions of roles and participation in distributed design processes (DDP), as linked to group or social awareness. It is focused on an open-source software community – the Python programming language community – as a model of DDP. On the basis of semi-structured interviews, we show that participants agree upon a typology of roles based on evident activities and experiences of participants, and that this knowledge guides their strategic use of archives for maintaining situation awareness. Contextualized interviews on a specific design process helps in understanding how this typology of roles is instantiated in a design situation and how social awareness is distributed among participants.

Introduction

Distributed design processes do not require that everyone within the project has a perfect and complete understanding of who is involved, for what tasks, with what experience or expertise at any time of the design process. It is also not required that

F. Barcellini (✉)
Ergonomics Lab, Research Center on Work and Development, Cnam, 41 rue Gay-Lussac, 75005, Paris, France
e-mail: flore.barcellini@cnam.fr

F. Détienne
Département SES – Bureau B 421, LTCI- UMR 5141 – CNRS – Telecom Paris Tech, 46 rue Barrault, 75634, Paris, Cedex 13, France
and
INRIA, Paris, France
e-mail: Francoise.detienne@telecom-paristech.fr

J.-M. Burkhardt
Ergonomics-Behavior-Interaction Lab, Paris Descartes University, 45 rue des Saints-Pères, 75006, Paris, France
e-mail: jean-marie.burkhardt@parisdescartes.fr

M. Lewkowicz et al. (eds.), *Proceedings of COOP 2010*, Computer Supported Cooperative Work, DOI 10.1007/978-1-84996-211-7_2,

among all the participants, one of them, e.g. the coordinator or project manager, has such knowledge available to support design problem solving and coordination decision. In fact, some minimal knowledge about the social community of participants could be enough, as far as this knowledge can support localization or mobilization of relevant agents at any step of the design process. An issue is to investigate content, forms, and ways of distribution of such knowledge among participants in a distributed design community.

The issue related to understanding how knowledge about participants in a distributed design community is constructed, distributed and mobilized as a resource for design, has been addressed under the heading of several theoretical frameworks in the literature. We will consider specifically the following concepts and their associated frameworks: social awareness [35] or group awareness [17] and team mental model [24]. Transversal to these theories is the (often ill-defined) notion of role, usually mobilized in other contexts of collective practices. It is consequently important to review these approaches in order to clarify both conceptual and methodological aspects relevant to investigate the issue of social awareness in the field of real distributed design communities. As emphasized by Prasolova-Forland and Diviniti [27], such knowledge can be considered as twofold: as a description of roles and participants within a community "in a general social situation", or as specifically contextualized knowledge related to specific events.

Our objective is to understand social awareness, and the mechanisms underlying the distribution of such social knowledge within participants in an Open Source Software (OSS) design community. OSS is probably one of the most representative cases of distributed design today. Although understanding how OSS communities are organized has been investigated recently through various approaches (e.g. [5,11–13,20, 21, 28, 29]), only very few studies have focused on group or social awareness in OSS (but see e.g. [17]). However, this issue has been recently recognized as most important for understanding the success of OSS.

In this paper, we focus on the Python OSS community dedicated to the design and use of the Python programming language, and on a specific design process in this community. After a review of the theoretical framework, we will present our research questions and methodological approach. Then, we will present the results and discuss the perspectives of this research.

Theoretical Framework

Distributed Knowledge Across Distributed Design Communities: Theoretical Approaches

Several theoretical approaches have addressed the issue of knowledge about participants in groups, such as teams, or in more distributed collective activities as found in online epistemic communities [10]. Whereas teams are composed of stable members,

often with identified statuses, online design communities such as OSS communities are composed of a large and unstable number of participants whose statuses and roles are constructed throughout interactions.

One theoretical approach is embraced by the notions of team mental model or shared mental model, both referring to an organized understanding of relevant knowledge that is shared by team members [24]. A team mental model is defined as team members' shared, organized understanding and mental representations of knowledge about key elements of the team's relevant environment, amongst which knowledge about roles and statutes.

Mostly applied to very minimal teams, sometimes reduced to dyads (e.g. [23]), this approach has several limits which have been emphasized in the literature. Firstly, the team mental model literature has overemphasized the overlapping perspective of sharing knowledge (knowledge common to all members) leaving aside more complementary or distributed perspectives [24]. Indeed, in a more distributed perspective, which may be more relevant for design online communities, members may be supposed to gain distinct knowledge depending on their activity. Furthermore this distribution of knowledge may be considered in a complementary perspective, as in the theory of transactive memory. According to this latter approach, the individual memories of group members are supplemented by shared knowledge of who knows what. Secondly, and in a related manner, the shared mental models approach leaves aside the social, cultural, and physical contexts within which mental structures are embedded [9]. It leaves aside norms, values, rewarding systems constructed and adopted in a social group. Finally, this approach is mostly static and de-contextualised from action.

The literature on awareness, and more particularly social awareness or group awareness, also addresses these questions of group knowledge related to roles with a more contextualised, socially and technically embedded perspective. Social awareness or group awareness is a lateral dimension of situation awareness [19], referring to practices through which participants regulate their activities to adapt to discrepancies and tasks interdependencies [19,20]. Social awareness is awareness of the social situation in a group or community in a shared environment, which can be physical, virtual or both.

Social awareness encompasses awareness of social situation in general and social situation at a certain moment. Prasolova-Forland and Diviniti [27] distinguish these two conceptions of social awareness in the literature, depending on the emphasis made on concrete events or on general concepts. According to the former approach (see e.g. [35]) social awareness is defined as awareness about the social situation of other people, i.e. what they are doing, whether they are engaged in a conversation and can be disturbed or not, and of who is around and what is up, at a certain moment of time. According to the latter approach, social awareness is considered in a broader context. An example is group-structural awareness (see e.g. [16]) as knowledge about people's roles, positions, status, responsibilities and group processes.

Prasolova-Forland and Diviniti's definition embeds both approaches considering awareness of social situation in general and social situation at a certain moment: people's roles, activities, positions, status, responsibilities, social connections and group processes.

Based on these various models, two main focuses seem to be as particularly relevant for exploring social awareness in distributed design communities: its distributed characteristic and its strong link with activity. Firstly, the distributed perspective could be explored in relationship with the distinction between general and specific social awareness. Secondly, the emphasis put on activity as an input and a main component of social awareness could also be approached through the notion of role, distinguishing one's own role and other participants' roles in a community. Our definition of role refers to effective and emerging behaviour of participants that appears and differentiates through interactions [1–3]. In online communities [22] and more specifically in OSS [17], the importance of constructing knowledge of other participants' roles has been stressed as required for getting involved in and participating in online interactions. We develop this notion of role in the following section.

Multiple Dimensions of Roles and Participation Profiles Essential for the Design Process

The role of a participant can be characterized by some degree of regularity in his/her activity in interaction with the group and technological artefacts [2,3]. Three types of roles are usually distinguished to characterize profiles of participation:

- *Interactive roles* identified through structural analyses of communication, e.g. level of participation and place in a network of communication.
- *Task-oriented roles* associated with both production and coordination activities: cognitive roles are oriented toward generation or evaluation of ideas; epistemic roles are oriented toward knowledge sharing activities [2]; coordination roles are oriented toward definition and reformulation of group objectives, or synthesis activities. These roles also refer to activities analyzed in previous studies of collaborative design (e.g. [6,11,25,26]).
- *Socio-relational* roles associated to activities whose objective is to facilitate interpersonal relationships, i.e. reducing conflict, harmonizing and researching consensus. They concern the creation of a free, expressive and participative context of work.

Thus the notion of role may be viewed and analyzed along several combined dimensions. For example, the epistemic dimension refers to the type of knowledge brought by participants; the interactive dimension refers to the activities of interaction management, e.g. opening of topics in discussions. Analyses along these primary dimensions allow a second-level analysis, where their combination allows participation profiles to be revealed (cf. [2]).

Up to now, two participation profiles have been identified in the literature as important for the performance of the design process:

- *Boundary spanners* who compensate for communications deficit between different groups [19,29,30,33]. They are literally persons who span the gap between

their organization and external ones [29,33]. Their activity is characterized by communication or behaviour between two or more networks or groups. They move across different teams transferring and translating information about the state of the project. This profile combines a task-oriented role focused on coordination and knowledge sharing with a socio-relational role focused on group support. This profile remains often invisible from the participants' perspective [15].

- *Leaders* who are persons recognized for their competences and to whom power is accorded. They have not forcedly the status of manager. They have a central interactive role characterized by high participation and highly quoted interventions. Their socio-relational role is oriented toward the creation of an harmonious work setting. Their task-oriented role is dominated by coordination activities.

Previous Studies on OSS Communities: From Trace-Based Analyses of Roles to Social Awareness

In OSS communities, roles and participation profiles have been mostly approached via traces-based analyses of interaction in three interaction spaces [6,29]: a discussion space (mailing lists, forums, chat), a documentation space (project web site, related websites, blog, wiki, online documentation), and an implementation space (source code and its development history). Several research works have aimed to identify roles which emerge in online communities (in particular, [4,18,34] for OSS; [14] for wikipedia and usenet). They have been mostly developed on the basis of analyses of traces oriented by researchers pre-categories. Their more or less explicit twofold objective is to understand the collective dynamics and to specify/construct external representations for supporting situation and social awareness in these communities.

Two types of analyses are usually conducted, structural analyses and social network analysis (SNA) (e.g. [20]), or, more rarely, content analyses based on more qualitative analyses of the content (e.g. [28]). Based on both types of analyses, we [6] have developed a methodological framework to identify roles along the interactive, cognitive, epistemic, coordination and socio-relational dimensions. We [4,6] have crossed structural analyses and message content analyses to identify profiles in the Python community, in particular the one of the boundary spanners. We shown that boundary spanners were characterized by specific interactive roles (in particular cross-participation in parallel same theme discussions in two mailing-lists), epistemic roles (transferring knowledge from application and computer sciences domains); and coordination and socio-relational roles. In online communities like usenet and wikipedia, Gleave et al. [14] have attempted to identify social role signatures on the basis of structural analyses, checking the convergence with roles identified on the basis of content analyses. Based on SNA, Sowe et al. [34] have identified knowledge brokers (or boundary spanners) as actors who participated across three mailing lists in the discussion space of the Debian OSS project.

Finally, most of these works have left aside the related question of understanding the knowledge constructed by members about others activities and roles in their community. This issue of group awareness has been addressed in a recent study by Gutwin et al. [16] in an OSS context. According to these authors, group awareness includes knowledge about who is on the project, where in the code they are working on, what they are doing, and what their plans are. In a perspective closed of Prasolova-Forland and Diviniti [27], they distinguish between general awareness of the entire team and more detailed knowledge of people that interviewed participants plan to work with. They found that social awareness is maintained primarily through text-based communication, e.g. strategies such as lurking on online discussions. Following this study, our objective is to understand social awareness in a DDP, the Python OSS community, and the distribution of such social knowledge within participants. As in Gutwin et al. [16], our methodology will be partly based on semi-structured interviews. In addition, we will also conduct contextualized interviews in order to address more properly situated knowledge of participants.

Research Questions

Techniques for analyzing interactions in OSS communities have been previously developed (e.g. [5,29]. These techniques can provide a referential view of argumentation exchange and process coordination within distributed design online communities. This view is constructed externally from the community, organizing and making visible a range of persisting traces of the whole process. Even, if such views provide a good help to understand what was going on in a particular space and at a moment of design (at least as far as traces and their treatment provide an accurate view of it), they don't convey much information about how social awareness is distributed and used to coordinate design among participants.

We would like to capture several dimensions of this specific area of design knowledge distributed within the community. This can be addressed in two directions. Firstly, social awareness can be thought as a generic –more or less shared and consistent across participants – mental model describing the participants and their participation within the design community. Secondly, each specific design process can be thought as a unique instance of such a model, that is both an entry and an output to the previous generic mental model. Looking comparatively at these two levels might provide an opportunity for identifying components, relationships and distribution of social awareness in distributed design communities.

We will adopt the notion of role to elicit this part of distributed social knowledge that refers to contributions associated to participants in the design process. Roles will be understood as knowledge about the emerging profiles of participants' contributions in the course of design.

Our research questions concern how social awareness is characterized, distributed and supported across members in distributed design. For that purpose, we propose to explore inter-subjective dimensions of roles, that is the way participants within the

community identify and categorize themselves and other participants' roles and contributions to the design process. As Prasolova-Forland and Diviniti [27] and Gutwin et al. [16], we investigate social awareness by distinguishing between general vs. situated and specific levels. More detailed exploratory issues concern:

- *Content*: To what extent do participants converge upon a typology of roles or profiles in the community and how do they characterize them? What cues are preferentially exploited by participants to characterize roles?
- *Distribution*: What is the scope of distinctive roles memorized by participants and do some roles remain invisible? Are there functional/operative views of roles depending on participant's own position in the design process?
- *Articulation Between Levels*: Can we distinguish between generic roles (instantiated only in particular design process) and personalized roles (permanent at the community level)?
- *Function*: How far is knowledge about others' roles useful and mobilized by strategies for maintaining situation awareness?

A Focus on the Python Community

Our research focuses on the Python project (www.python.org) which is a dynamic and stable project. Created in 1994, this project has a group of about 100 designers and a lot of potential users of Python in various application domains (web design, scientific computing, gaming, finance…).

An interesting characteristic of this project is that its participants can engage in a specific design process: the *Python Enhancement Proposal* process (PEP). This process is used by the Python community to frame proposals of new functionalities and evolutions of the Python language.[1] Everyone can propose a PEP, whatever his/her status; the proponent is called the champion. PEPs are then discussed in two main mailing-lists of the project (use and design oriented mailing-lists). Focusing on PEPs processes enables us to select contextualised online data, in particular data concerning specific design activities.

As in other OSS projects, the design process is distributed among the three online interaction spaces [29]: the discussion, documentation and implementation spaces.

The discussion space of the Python project is composed by two main mailing lists:

- A use-oriented mailing list, *python-list*, is about general discussions and questions about Python.
- A design-oriented mailing list, *python-dev*, is for work on developing Python (fixing bugs and adding new features to the Python language).

[1]The PEP process is similar to other design processes in distant design communities, like the Request for Comment used by the Internet Engineering Task Force, the XEP process used by the jabber community (www.jabber.org). This formal process is also closed to the consensus-based decision making of Apache.

The documentation space is composed by all documents, website, blogs, wikis relative to the project. It contains all PEP's documents and their versions.

The implementation space contains the proper code of Python: the standard library and its modules, their versions. All the code versions related to a new functionality proposed by a PEP are archived there.

Methodology

Our methodology combines two types of interviews, each one targeting one of our principal research questions.

Our research questions concerning general social awareness are addressed by conducting semi-structured interviews with participants in the Python community.

Our research questions concerning situated social awareness are addressed by conducting semi-structured contextualized interviews with participants in a particular design process episode. The chosen design process concerns PEP XYZ.[2] Previous studies based on analyses of traces [4,6] allowed us to identify the scope and forms of participation in this PEP design process.

This PEP process occurred between October 2003 and May 2006. As our interviews were performed *a posteriori* (during 2007), this might be seen as a limitation for accessing for participants' representation in context. However, a previous work on memory of interactions in cooperation [8] has outlined that, with a long delay between the recalled interaction and the interviews, verbal content is massively forgotten but interactional and relational positions are still well remembered by interviewees.

Semi-Structured Interviews

Participants Recruitment

To recruit participants, we contacted chairs of the Python French Association (www.afpy.org) and we recruited interviewees during two Python's conferences (World Meeting on OSS – *Rencontres Mondiales du Logiciel Libre* – and Europython'06). Each interviewee was asked to recommend any other relevant participants to the researcher. We also contacted participants on the basis of observations of the mailing-lists of the Python project.

We recruited 14 interviewees. They were members of the Python community in France, United-States, Argentina and Australia. Statuses of participants in the Python community and the modalities used to conduct interviews are synthesized in Table 1. Interviewees are all users of Python, some are developers of the proper

[2]This PEP concerns the introduction of a module for decimal calculation, pushed by users in the financial domain.

Table 1 Interviewees statuses and modality used to conduct interviews

Statuses	Number	Modalities
Developers in application domains	5	Face to face
Developers in application domains	5	Phone
Project leader	1	Face to face
Developers of Python language	3	E-mails

Python language and some are developers of software based on Python in various application domains (web, scientific computing, financial).

Collected Data and Analysis

Face to face interviews consisted in 1–2 h of exchanges. E-mails interviews are composed of three to four messages. The interviews have been transcribed and form a corpus composed of 5,500 lines of text.

We performed a thematic qualitative analysis. Themes were categorized according to:

- Characterization of interviewee's role and participation in the project (participation in one or another mailing-lists, code contribution …)
- Types of roles and criteria reported by participants (project leader, area of expertise…)
- Strategies for constructing and maintaining knowledge about roles and design processes (reading mailing-lists, blogs or wikis, subscribing to news or RSS feeds…)

Contextualized Interviews Concerning a Design Episode

Interviewees' Recruitment and Resulting Sample

One hundred and twenty participants were involved in the use-oriented and design-oriented mailing-lists concerning the PEP XYZ process. Among them, we identified 14 common participants on both lists, from which five could be characterized as boundary spanners (for more details, see [4,6]). Seven participants contributed in the documentation and the implementation by modifying the source code and/or the PEP specification document. Actually, two participants acted in the documentation space, six in the implementation space, and one in both spaces. Thus, the technical profile related to implementation of code is concentrated in a smaller group (7) than the discussant profile (120 of which 24 are high discussants).

Among these participants, we requested 16 of the high discussants to perform an interview by email. Only four participants accepted to answer and three of them followed all the interview process. Nevertheless, the three interviewed participants are characterized by specific and complementary profiles of participation reconstructed on the basis of our previous studies [4,6] and the semi-structured interviews presented above.

Two interviewees acted in both discussion and implementation spaces:

- *Fabio*[3] is the champion of the PEP that appears to have a profile of leader during the design process characterized by a central position in communication and a tendency to perform coordination activities. He is also one of the five boundary spanners identified.
- *Nelson* was one of the most frequent participants to discussions during the design process. He is also one of the six participants who modified the source code corresponding to the PEP.

The third interviewee, *Anthony*, acted only in the discussion space. He is a well-known expert of the project (one of those identified thanks to the semi-structured interviews). He is working in the same organization than the project leader. He appears to be also one of the five boundary spanners of the design process.

Collected Data and Analysis

The questions addressed by the interviewees concerned:

- The history of the PEP XYZ process
- The participation of the interviewee in this process, his/her potential influence and his/her motivation
- The participation of others, their roles in the process, their potential influence and their motivation.

In a first phase, interviewees were asked to answer questions without looking at the PEP discussions archived online. In a second phase, in which only two interviewees (Fabio and Anthony) accepted to participate, they were allowed to consult archives.

The collected email interviews represent 1,000 lines of text.

We performed a qualitative content analysis of interviews by extracting and categorizing the descriptors of roles, and the name of participants remembered by interviewees. We analyzed both the knowledge of participants about their own participation, and the participation of other recalled people. In the following, only the results relevant for our research questions will be presented.

Results

This section is organized according to results obtained from the two types of interviews, each one targeting one of our principal research questions.

Concerning our research questions on general social awareness, semi-structured interviews leads us to identify a typology of roles within the Python Project, and strategies developed by participants for constructing and maintaining awareness.

[3]Faked names.

Concerning research questions on situated social awareness, semi-structured contextualized interviews with participants in PEP XYZ process allow us to specify the representations developed by participants about the roles of key actors in this process.

Typology of Roles and Strategies for Maintaining Awareness

The typology of roles presented in this section is based on an agreement between participants about dimensions they employ to identify roles at the community level and refer to general social awareness. We stress one main disagreement on the definition of a shared understanding of the roles of users and developers. We conclude this section by presenting strategies employed by participants to elaborate and use this knowledge for maintaining situation awareness.

Underlying Dimensions of Roles and Profiles

At a general level, participants elicited a partially shared view on participants' roles within the OSS community. Their knowledge about roles in their community is built on the basis of several principles. Globally, participants agreed upon three types of principles for identifying roles.

The first principle used for distinguishing roles is based on participants activities in the interaction spaces. Indeed, all participants described spontaneously different types of participants according to their activity in one or another spaces:

- *Discussants* profiles (related to evidences of activity in the discussion space of the community), are defined by participation to mailing-lists of Python (use-oriented, design-oriented or application domain) and sometimes by explicit reference to the quality of participation, as pointed by a participant: *"They are some people that construct so well-constructed their answers (in online discussions" that they are call "'bot' a robot: who write, scan all that is written online and automatically send good answers"*.
- *Technical* profiles (related to evidences in the documentation and implementation spaces), are defined by software design actions (bug report, selection of bug, providing correction of bug, verifying quality of code, documentation, managing code versions…).
- *Outside of interactions spaces* persons related to activities oriented to evangelization and promotion of Python through book writing, organization of conference etc…

Furthermore, interviewees converged in eliciting one particular *discussant profile*, linked to *boundary spanning* activities [29]. Mediation and interpersonal issues are evoked and related to information transmission, support of other participants in proposing a new functionality.

The second principle is linked to reputation and expertise, as stressed by a participant:

> I think that there is no formal role with a tag. I think that they are simply people that are very visible because they contributed a lot and who have particular expertise.

These roles are most often personalized. Participants referred directly to the person name, his/her expertise domains and sometimes the context in which the person developed his/her expertise.

> For instance, when one talks about, for instance, numeric…it is always Tom…because Tom (…) is a deep expert in scientific computing. Tom worked for a while in a big scientific computing institution. Thus, if one has a problem with that if Tom says « that » that's it. It is not a formal role, it is just that he has some skills.

These roles are similar to the notion of *super-experts* or gurus as defined in the literature about software design [32]. From the interviews, we obtained a group of nine participants (identified by their names) seen as super-experts. Interestingly, the fact that five of them were administrators and four were developers of Python suggests that such role is not systematically associated to the participants' status.

The third principles is linked to the participant's status. This principle is applied only for the project leader (and not for administrators/developers/users as we will discuss in next section). The most personalised role on which every participants agree upon was the *project leader*. This is interestingly the only profile which is confounded with both a particular person and a recognized status within the community. Indeed, the project leader of Python is referred to by his status of project leader *the "Benevolent Dictator For Life" (BDFL) with veto rights over anything* but also according to his skills and his decision making activities referring to an effective profile of leader. Thus, the status of project leader of Python corresponds to his activities*:* his skills are recognized and power is accorded to him.

The Distinction Between Developers and Users Do Not Correspond to a Shared Understanding of Roles

Distinction between users and developers is made in the epistemology of design and in design process models. In the OSS context, these two terms keep on being used in two different ways:

- To distinguish formally users and developers' statuses in reference to ownership, and particular rights to modify the source code
- To distinguish activities in reference of a particular designed artefact (e.g. use or design of the python language)

On the basis of our interviews, it seems that there is no agreement between participants on the meaning of the developers term: some interviewees refer to discussant activities in the design-oriented mailing-lists, other to technical contributions, other to rights to modify the code, and the project leader defined developers are those who contribute significantly to the implementation of Python. Thus, as one of the interviewee put it:

> [Developer's] a vague term which may in different contexts be used more or less restrictively.

Things are also made unclear by the fact that users and developers definitions are relative to the context: for instance one can be a user of the Python language and a developer in an application domain.

> Python's users are developers that use Python to create products that will have their own users. They are not language developers they are developers that are using the language. Then, I feel I belong the users' community of Python, but not to the community of people that make the language evolved.

Finally these results question the classical distinction between users and developers (or designers) made in the epistemology of design.

Strategies for Constructing Knowledge About the Python Community

Two kinds of strategies are revealed by interviewees to construct social awareness, but also situation awareness (for instance what is the state of development of the project?): one is based on direct experience of participants in the three interactions spaces, another is based on mediation by online resources such as blogs or wikis. These two kinds of strategy are strongly linked to online interaction spaces of the project, in particular the discussion space.

Strategies based on experiences of participants are related to:

- Their participation in online discussions as discussants; for instance, some interviewees could not always answer to questions about roles because they did not have a look on mailing-lists for a while
- Their participation in the implementation space
- Their use of the code, as the name of the contributors is specified in the code for instance

Strategies based on mediation are related to:

- Automatically generated online resources, such as news on the website of the project or digests of online discussions

> So, when that [new functionality] makes a lot of noise, that come out on Python dayly news…
>
> On the website I go on ' what's new ?' that way I can follow evolutions
>
> I've got also a google alert, using keywords

- Resources constructed and made publicly visible by others participants: such as blogs of project leader and super-experts, or wikis. This strategy is thus strongly related to the knowledge about roles in the community which has been constructed by the participants. As in the transactive memory approach, participants use their knowledge on "who knows what", as a pointer to relevant resources:

> what makes the difference between a buddy who knows and one who doesn't know, is the buddy who spent two years polishing his sources…And then, once you know important people in the community, you point his blog and that's it.

Representations of Key Actors' Roles in a Specific Design Process

Results presented in this section are based on the contextualized semi-structured interviews about PEP XYZ process. Firstly, they concern the definition of a typology of descriptors used by interviewees to evoke roles. Secondly, they show the scope of participants recalled by interviewees. Finally, we outline that interviewees have different perspectives about roles they evoked, and that some roles can remain more or less visible for participants.

Underlying Dimensions of Contextualised Roles and Profiles

The following table (Table 2) presents a categorization of descriptors used by interviewees to evoke roles. This categorization is based on the distinction between discussant and technical profiles, and refined by more specific roles for the discussant profile [4,6].

Table 2 Interviewees' descriptors of roles

Profiles	Roles	Descriptors	Excerpt from interviews
Discussant profile	Interactive	Regularity of answers to posted messages	There were a lot of mails to the lists […]. I read them [all] at home […] preparing answers
	Epistemic	Usefulness of provided knowledge	[They helped me] with knowledge
	Cognitive	Relevance of answers	X offered useful comments
		Initiation of design idea and proposals of solution	Y was the source of the idea
	Coordination	Compilation of group input	X did all the work […] gathering feedback, incorporating it
		Consensus building	Make all this people to accord in a unique result
	Socio-relational	Mentoring help	X basically mentored Y through the process of writing the PEP
			X came and said: "you should first build the decimal data type, take a look at here and here"
			X, Y helped a lot with all this
Technical profile	Code production	Participation to pre-code or code implementation	Just the start of it [code], with a little of tests
		Coding of test	A lot of work cleaning the code and in documentation
		Participation to documentation issues	X who had a semi-worked out module […] Y who had done further work on X's module
		Code optimization	
		Reference to code ownership	

For the interactive role, participants refer to the regularity of presence in discussions, and the recurrence of answers.

For the cognitive and epistemic roles, participants refer to other participants' input in terms of usefulness of knowledge with respect to the design or relevance of the input in the discussions. They do not refer to content and type of knowledge (e.g. (application domain versus computer science or language domains) brought out by participants in the discussions. Their characterisation of roles is rather functional and normative. These characterizations are sometime embedded in a temporal description of the design process, whereas the content of the contributions remain fuzzy. Anthony, for instance, said:

> When the PEP was opened (Jan 2004) Georges gave some encouragement […] then Michel offered useful comments; in the following month (Feb 2004) useful input was again offered by Aziz and Tom.

The coordination role, considered as most important in research on group work (e.g. [23]), is referred to only by the champion and his description remains very general.

> The rules to get a PEP accepted are simply: You have to […] dive into thousand of mails of hundreds of people proposing absolutely useful ideas […], read them, answer them, make all this people to accord in a unique result.

For characterising the code production roles, interviewees' descriptors may be quite precise and detailed. The participants distinguished between contributions concerning the pre-code, the code, tests, code optimization and the PEP document. Furthermore they also considered code ownership (X's module). This notion of ownership is also evoked in the cognitive role viewed by interviewees when they mention that "Y was the source of the idea".

Scope of Participants Evoked by Interviewees

The scope of identified roles, after the first phase of interviews, varies from three, for the champion (Fabio), to only one or two, for the other interviewees. The third interviewee (Anthony), who performed the second phase of interview, in which he was allowed to explore the archives of the PEP, extended a lot his scope of discussants from two to ten. Figure 1 illustrates the scope of participants evoked by the three interviewees.

Combining Fabio, Nelson and Anthony's extended views leads us to obtain a group of:

- Two participants recalled by all interviewee (Fabio and Robert)
- Three additional participants recalled by Fabio and Anthony (Tom, Aziz and Ernesto[4])

[4]Ernesto was not anymore part of the project at the time of the PEP, but was the owner of the first code, with Aziz, to implement decimal in Python. In this sense, it is referred by participants to be a contributor in respect of a code ownership value in OSS communities. For this reason, we do not discuss any further his profile.

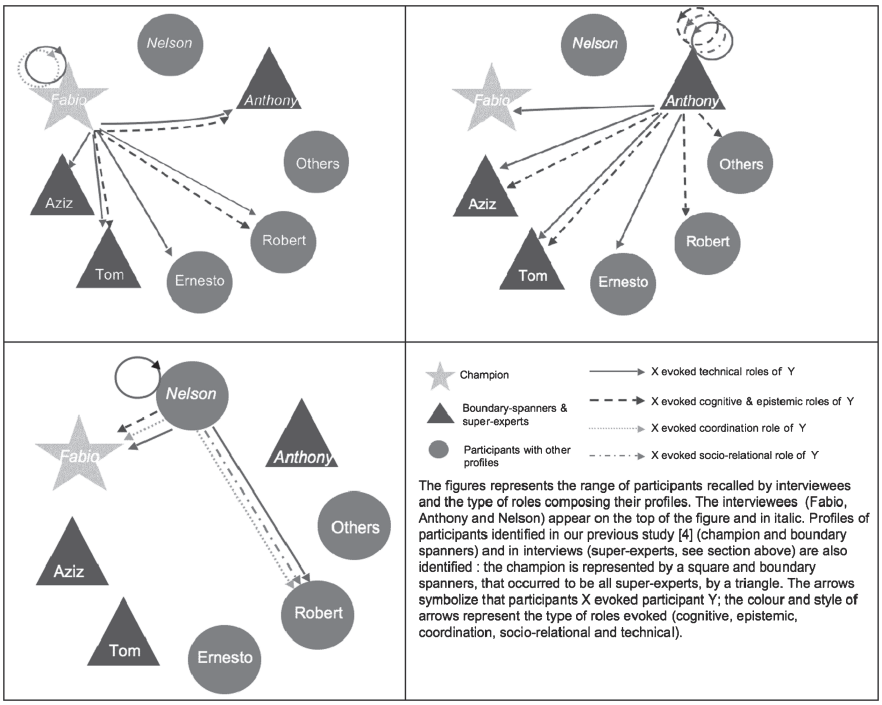

Fig. 1 Scope of participants' roles and perspectives on roles evoked by interviewees

- Four other participants recalled only by Anthony

It is interesting to note that after the first phase of interviews (without any traces support), Fabio (the champion) is the one that has the more extended view.

In the following we focus on participants recalled by at least two interviewees as presented in the figure above (Anthony, Fabio, Robert, Tim, Aziz and Ernesto), in order to compare their perspectives on roles of participants.

Interviewees' Different Perspectives on Roles

The perspectives of participants on one or another roles seem to depend on their own participation in the interaction spaces of the project.

The case of Robert is interesting as he is the only participant recalled by all three interviewees. Fabio, Anthony and Nelson agreed on his important role, but they did not characterize in the same way both his discussant and technical profiles (Fig. 1):

- Concerning Robert's technical profile, both Fabio and Nelson described it as essential. For Fabio, Robert did "*a lot of work in cleaning the code and documentation*" and for Nelson, Robert "*was the one who made all the [code*

modification] in the repository". Indeed, Robert was the main code contributor (nearly 80% of actions in this code) [6]. He is an administrator of the project but is not recognized as a super-expert in our interviews. However, Anthony did not refer to the contributions of Robert in coding and documentation

- Concerning Robert's discussant profile, the three interviewees had different perspectives. Fabio acknowledged the importance of the knowledge brought by Robert. Anthony referred to it only as useful inputs. But the most interesting difference remains in Nelson's view that makes apparent a strong implication of Robert in coordinating the design process and mentoring the champion:

  Robert provided strong support to the idea as an established participant in the Python project, and basically mentored Fabio through the process of writing the PEP, submitting it and getting it accepted.

Thus, only Fabio and Nelson who were present in both the discussion and technical spaces have a broader perspective of his profile. These results seem to indicate that participants' perspectives on others' role is linked to their own implication in the spaces. Anthony who was only a discussant in the process may not be aware of the huge impact of Robert in the code implementation as himself did not participate in the technical side. By contrast, Fabio and Nelson who were involved in both spaces refer to the importance of Robert's technical profile, and their views are complementary.

A Relative Invisibility of Boundary Spanners and a Strong Visibility of Technical Profiles

Our interviewees recalled four out of the five boundary spanners that we identified in our previous research [4]: Fabio, Anthony, Tom and Aziz (Fig. 1). This result may indicate that their key participation during the design process [6] has been stressed by interviewees. Participants appearing to be boundary spanners correspond to more than half of participants evoked by Fabio and Anthony. However, if interviewees agreed upon the technical profiles of these participants, they diverged on their discussant profiles, minimizing their socio-relational and epistemic contributions.

Concerning Tom and Aziz, both interviewees recognized that they contributed to pre-code (initial versions of code) revealing their technical profile. Tom was the technical expert of the solution used to do financial calculation with Python before the PEP started. Aziz was the owner of one first attempt to implement the function proposed by the PEP. Concerning their discussant profiles, Fabio refers to the knowledge they brought ("*A lot of knowledge*") whereas Anthony spoke only of their "*useful inputs*" at the same level as for other participants.

The same type of results can be found concerning Fabio (the champion), Anthony and Nelson agreed upon Fabio's technical profiles but not on his discussant profile. Anthony emphasized the contribution of Fabio on the technical side, and

did not refer to his boundary spanner and leader profiles, whereas it was essential for Fabio and Nelson. As put by Fabio himself:

> You have to dive into thousands of mails of hundreds of people proposing useful ideas and also absolutely shitty ones, read them, answer them, make all this people to accord in a unique result

On the same way, Fabio did not refer to the strong support provided by Anthony, as he noted himself and as we observed in our previous studies [4,6]. Fabio emphasized the cognitive and epistemic roles of Anthony and did not stress his socio-relational role.

These results may reflect the tendency to emphasize technical contributions in OSS communities, and to "forget" socio-relational aspects as essential for the performance of the design process. Even though participants acting as boundary spanners are recognized as key participants on the technical and sometimes on the cognitive and epistemic sides, their broader profile of boundary spanners is not acknowledged by interviewees. As a matter of fact, it is interesting to note that we showed in our previous trace-based analyses [6] that the presence of boundary spanners and socio-relational activities, performed by Fabio, Anthony and Robert, may be one of the key element explaining the success of the PEP proposal.

Discussion and Perspectives

These results help in understanding the content, development and distribution of social awareness within a large and somehow fuzzy bounded online OSS community. They reveal several dimensions of social awareness articulated across generic or personalised roles and un-contextualized or contextualized roles. Furthermore, these roles are differentially recognized by participants depending on the participant's own perspective and activities, as well as their capability in collecting relevant information.

Concerning the content, we showed that participants seem to agree upon a typology of generic roles based on evident activities performed by participants in the interaction spaces of the project and on personalized roles based on reputation and expertise of participants. They agree that the status is not relevant as a principle to distinguish between roles. Indeed, the distinction between users and developers does not capture the multiple dimensions of participation performed in design activities.

This content about general and personalized roles can be assimilated to a general mental model of participation in OSS communities. This global level of social awareness is developed and maintained by direct experience of participants within the interactions spaces of the community, as well as by gathering information through Internet resources such as blogs of participants recognized as resources for others. The knowledge about these key actors of the project would be an element supporting the transactive memory of participants [24].

Concerning the content of social awareness related to specific episodes of design within the community, our results make clear the distribution of social awareness between participants. We showed that none of the participants has an exhaustive perspective of roles. This distribution is outlined by the difference in scope of roles

evoked by interviewees, as well as by the difference in their perspectives on these roles. This difference appears to be linked to their own position and activity of the interviewed participants in the design process. These various perspectives may be considered as entries for an extended social awareness as they embrace parts of a network of participants being resource for transactive memory construction.

It is interesting to discuss also the articulation between global and situated social awareness and the way it is shared or distributed across participants. Participants evoked personalized super-experts profiles, generic profiles of boundary spanner, discussant and technical profiles at a generic level of description of their community. Some of these roles or profiles are more precisely instantiated in the context of a specific design process episode in relation to evident activities of participants. We show that this instantiation is partial. Firstly, there is an emphasis on technical contribution of participants, reflecting a shared value in OSS communities regarding code production. Secondly, socio-relational and support activities – two grounds of the boundary spanning profile – are not well acknowledged by participants in design episode, whereas mediation and boundary spanning generic roles are recognized by participants at the global level. We showed in a previous study that the presence of boundary spanners was a key element of the success of the design process we studied [6]. In the same direction, literature on collaborative activities emphasizes the importance of boundary spanning and coordination in the performance of group work, suggesting also that boundary spanners are not necessarily recognized by the group [19,23]).

These results suggest that both community and team levels (seen as an *ad hoc* community subset collaborating in a specific design episode) are components of distributed social awareness in OSS design. Participants share (1) generic roles that could or should be taken in the design processes; (2) specific person-roles coupling which point on leaders or super-experts in the community; (3) detailed (although filtered) evidence-based perspectives of clearly identified participants' roles based on direct interactions during episodes of design.

This research extends previous works on group awareness and roles in online communities [17], pointing out the importance of social awareness in the elaboration of resources for participants. We propose an analysis of content, clues and dimensions of social awareness as constructed by participants in distributed design in an OSS community. Our results provide also insights on the strategies underlying the construction of awareness, such as the importance of knowledge of roles in order to select resources for maintaining situation awareness. Finally, our findings reveal that memorization of social awareness is a robust mechanism as its constituting elements (roles) can be evoked several years after a design episode. More detailed study of these strategies and how they affect the construction and maintenance of social awareness should be carried out.

Our perspectives of research concern social awareness and reflective tools or methods to support group activities [7]. In this direction, an issue is to investigate to what extent strategies developed by participants to construct situation and social awareness are efficient and how reflective tools can be incorporated in activities. Our position is that a role-based view of the communities could be interesting at least for new comers and also as a support for reflective activities on the collective

dynamics. An open issue concerns the effect of revealing invisible roles or profiles on group regulation. A second phase of interviews is under progress, in which we provide the champion with our graphical representations of traces-based profiles in the PEP process.

Acknowledgements We thank our interviewees for the time they have spent to answer our questions.

References

1. d'Astous, P., Détienne, F., Visser, W., and Robillard, P. N. (2004). Changing our view on design evaluation meetings methodology: a study of software technical evaluation meetings. *Design Studies, 25,* 625–655.
2. Baker, M., Détienne, F., Lund, K., & Séjourné, A. (2009) Etude des profils interactifs dans la conception collective en architecture. In F. Détienne, V. Traverso (Eds) *Méthodologies d'analyse de situations coopératives de conception: le corpus MOSAIC*, Nancy : PUNBales, R.F. (1950). *Interaction process analysis : a method for the study of small groups*. Cambridge : Addison-Wesley.
3. Bales, R.F. (1950). *Interaction process analysis: a method for the study of small groups*. Cambridge : Addison-Wesley.
4. Barcellini, F., Détienne, F., and Burkhardt, J.M. (2008a). Users and developers mediation in an Open Source Software Community: boundary spanning through cross participation in online discussions. *International Journal of Human Computer Studies, 66(7),* 558–570.
5. Barcellini, F., Détienne, F., Burkhardt, J.M., and Sack W. (2008b). A socio-cognitive analysis of online design discussions in an Open Source Software community. *Interacting with computers, 20*, 141–165.
6. Barcellini, F., Détienne, F., and Burkhardt, J.M. (2009). Participation in online interactions spaces: design-use mediation in an Open Source Software community. *International Journal of Industrial Ergonomics, 39*, 533–540.
7. Bodker, S. and Christiansen, E. (2006) Computer Support for Social Awareness in Flexible Work. Journal of Computer Supported Cooperative Work, *15* (1), 1–28.
8. Cahour B. (2002). How the subjective memory of interactions at work makes cooperation complex. *Revue des Sciences et Technologies de l'Information, série RIA, 16* (4–5), Numéro spécial "*Cooperation and complexity in sociotechnical systems*".
9. Carroll, J.M., Rosson, M.B., Convertino, G., and Ganoe, C.H. (2006). Awareness and teamwork in computer-supported collaborations. *Interacting with Computers 18*, 21–46.
10. Cohendet, P., Creplet, F. and Dupouët, O (2000). Organizational innovation, communities of practice and epistemic communities: the case of Linux. In A Kirman & JB Zimmermann (Eds) *Economics with Heterogeneous Interacting agents*. The Netherlands: Springer.
11. Détienne, F., Boujut, J-F., and Hohmann, B. (2004) Characterization of Collaborative Design and Interaction Management Activities in a Distant Engineering Design Situation. In F. Darses, R.. Dieng, C. Simone, M. Zaklad (Eds) *Cooperative Systems design*. IOS Press, 83–98.
12. Ducheneaut, N. (2005). Socialization in an Open Source Software Community: A Socio-Technical Analysis. *Journal of Computer Supported Collaborative Work, 14*, 323–368.
13. Gacek, C., and Arief, B. (2004). The Many Meanings of Open Source. *IEEE Software, 21*, 34–40.
14. Gleave, E., Welser, H.T., Lento, T.M., and Smith, M.A. (2008). A Conceptual and Operational Definition of 'Social Role' in Online Community. *42nd Hawaii International Conference on System Sciences*, 2009, pp. 1–11.

15. Grinter, R. (1999). Systems architecture: product designing and social engineering. In Proceedings of the *international joint conference on Work activities coordination and collaboration*, pp. 11–18. ACM Press.
16. Gutwin, C., Greenberg, S., and Roseman, M. (1996). WorkspaceAwareness in Real-Time Distributed Groupware: Framework,Widgets and Evaluation. *In proceedings of HCI 1996.*
17. Gutwin, C., Penner, R., and Schneider, K. (2004) Group Awareness in Distributed Software Development. In *Proceedings of CSCW 2004* (pp. 72–81). New York, USA: ACM press.
18. Hendry, D.G. (2008) Public participation in proprietary software development through user roles and discourse. *International Journal of Human-Computer Studies, 66,* 545–557.
19. Krasner, H., Curtis, B., and Iscoe, N. (1987). Communication breakdowns and boundary spanning activities on large programming projects. In G. Olson, S. Sheppard, and E. Soloway, E. (Eds.) *Empirical Studies of programmers: Second Workshop*, pp. 47–64.
20. Lopez-Fernandez, L., Robles, G., Gonzalez-Barahona, J.M. (2004). Applying social network analysis to the information in CVS repository. In *International Workshop on Mining Software Repositories*, Edinburgh, Scotland, 25th May.
21. Mahendran, D. (2002). *Serpents and Primitives: An ethnographic excursion into an Open Source community.* Master's Thesis, University of California at Berkeley.
22. Maloney-Krichnar, D., and Preece, J. (2002). The Meaning of an online health community in the lives of its members: Roles, relationship and group dynamics. In *Proceedings of the 2002 International Symposium on Technology and Society ISTAS'02*, Social Implication of Information and Communication technology, 20–27.
23. Mathieu, JE., Heffner, TS., Goodwin, GF., Salas, E., Cannon-Bowers (2000). The influence of shared mental models on team process and performance. *Journal of Applied Psychology, 85*, 273–283.
24. Mohammed, S. and Dumville, B.C. (2001). Team mental models in a team knowledge framework: expanding theory and measurement across disciplinary boundaries. *Journal of Organizational Behavior, 22*, 89–106.
25. Olson, G.M., Olson, J.S., Carter, M.R., and Storrosten, M. (1992). Small Group Design Meetings: An Analysis of Collaboration. *Human-Computer Interaction, 7*, 347–374.
26. Olson, G. M., and Olson, J. S. (2000). Distance Matters. *Human-Computer Interaction, 15*, 139–178.
27. Prasolova-Forland, E., and Divitini, M. (2003). Supporting Social Awareness: Requirements for Educational CVE. *Third IEEE International Conference on Advanced Learning Technologies (ICALT'03)* (pp. 366).
28. Ripoche, G. and Sansonnet, J.-P. (2006). Experiences in Automating the Analysis of Linguistic Interactions for the Study of Distributed Collectives. *JCSCW, 15(2–3),* 149–183.
29. Sack, W., Détienne, F., Ducheneaut, N., Burkhardt, J-M., Mahendran, D., and Barcellini,F., (2006) A methodological framework for socio-cognitive analyses of collaborative design of Open Source Software. Journal of *Computer Supported Cooperative Work, 15(2–3)*, 229–250.
30. Sarant, S.A. (2004). *The role of organizational boundary spanners in industry/university collaborative relationship*. Doctor of Philosophy in Psychology Dissertation Thesis. North Carolina State University, 2004.
31. Schmidt, K. (2002). The problem with 'awareness': introductory remarks on 'awareness in CSCW'. *Journal of Computer Supported Cooperative Work, 11*(3–4), 285–298.
32. Sonnentag, S. (1998) Expertise in professional software design: A process study. *Journal of applied psychology, 83(5),* 703–715.
33. Sonnenwald, D.H. (1996). Communication role that support collaboration during the design process. *Design Studies, 17*, 277–301.
34. Sowe, S. Stamelos, I. Angelis, L. (2006). Identifying knowledge brokers that yield software engineering knowledge in OSS projects. *Information and Software Technology, 48,* 1025–1033.
35. Tollmar, K., Sandor, O., Schomer, A. (1996). Supporting SocialAwareness. @Work Design and Experience. In *proceedings of CSCW' 1996* (pp. 298–307).